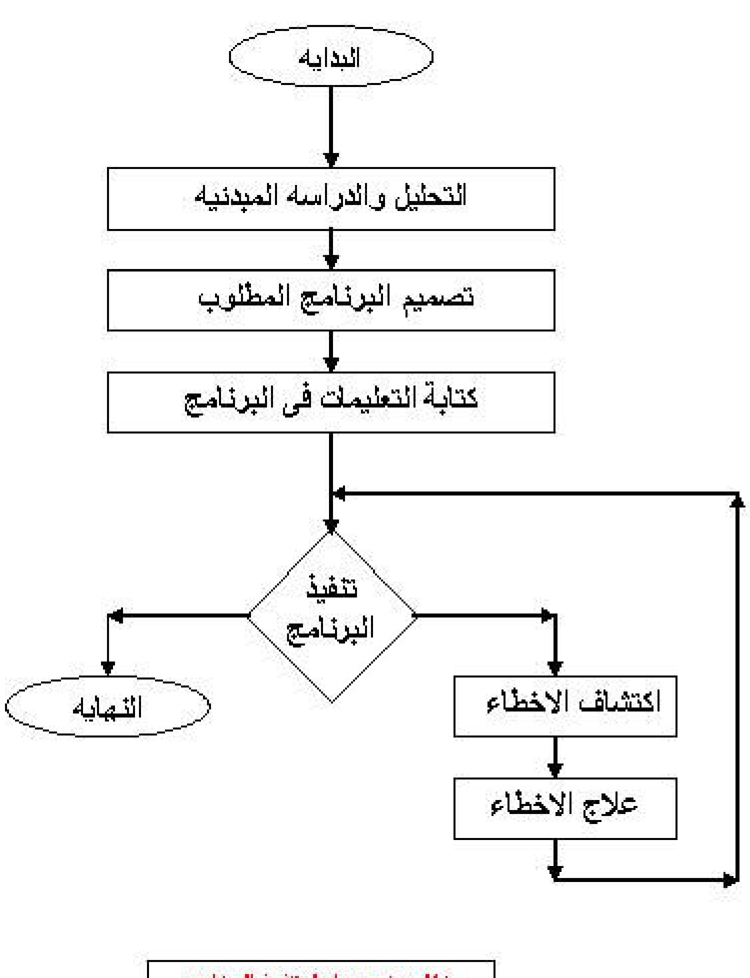


تعقب وتصحيح الاخطاء فى الدوت نت

بسم الله الرحمن الرحيم
تعقب وتصحيح الاخطاء فى الدوت نت

عند الشروع فى اعداد برنـامج معين لحل مشكله معينـه يتم التعرف اولا على نـوع المشكله والقيام بعمل الدراسه المبدئيه للمشكله ثم تصميم البرنـامج الذى يقوم بحل هذه المشكله ثم كتابة تعليمات البرنـامج واختبـاره وفى حالـة وجود اخطاء فلا بد من معالجتها ، وكما ترى فى الشكل التالى



شكل يوضح مراحل تنفيذ البرنامج

ان اكتشاف الاخطاء وعلاجها داخل البرنـامج يعتبر عنصر هام فى دورة اعداد اى برنـامج وهى خطوه مطلوبـه للوصول الى برنـامج نـاجح وذالك لان اى مبرمج مهما كانت خبرته قد يقع فى خطأ اثنـاء كتابـة البرنـامج لـذلك عليـه ان يتعرف على امكانيـات اكتشاف الاخطاء المختلفه والطرق السريعه للعلاج لتوفير الوقت والجهد .

وبم ان البرنـامج يحتوى على اخطاء فى كل الاحوال فيصبح من الضرورى العثور على هذه الاخطاء واللتى تدعى **Bugs** ومن ثم تصحيحها، يمدك الفيجوال بيسيك بكثير من الادوات التى تساعدك فى ذالك اذ يشتمل على بيئـة تنقيح كاملـه تحتوى هذه البيئـة على:-

١. مراجعه هجائيه لتتاكد من انك كتبت الاوامر بصوره صحيحه .
٢. رؤيه انيه للمتغيرات **Watching Variables** لترى قيم المتغيرات اثناء تشغيل البرنامج .
٣. تعقب الكود **Code Tracing** يجعلك تعرف السطر الذى ينفذه المترجم وتأثير تنفيذه .
٤. سرد للاجراءات قيد النداء **Listing Call Procedure** لتعرف من اى الاجراءات تم نداء الدوال او الاجراءات الحاليه .

وقبل ان تعرض بالشرح لطرق تتبـع الاخطاء وتصحيحها سنتعرف اولا على انـواع الاخطـاء التـى يمكن ان يقـع فيهـا المبـرمج وكيفية تلافيها :

انواع الاخطاء

يمكن تقسيم الاخطاء التى يقع فيها المبرمج الى ثلاثة انواع :

- اخطاء هجائيه Syntax Errors
- اخطاء اثناء التنفيذ Run-Time Errors
- اخطاء منطقيه Logical Errors

اولا : الاخطاء الهجائيه

تنتج هذه الاخطاء عند كتابة تعليمات لا تراعى فيها قواعد اللغه المستخدمه للبرمجه عموما ،ويقوم محرر الفيجوال بيسيك(نافذة الكود) بالتعرف على بعض الاخطاء من هذا النوع بمجرد كتابة التعليمات بينما يتم التعرف على باقى هذه الاخطاء فى مرحلة تنفيذ البرنامج ومن امثلة هذا النوع من الاخطاء الخطأ فى كتابة امر داخل البرنامج :

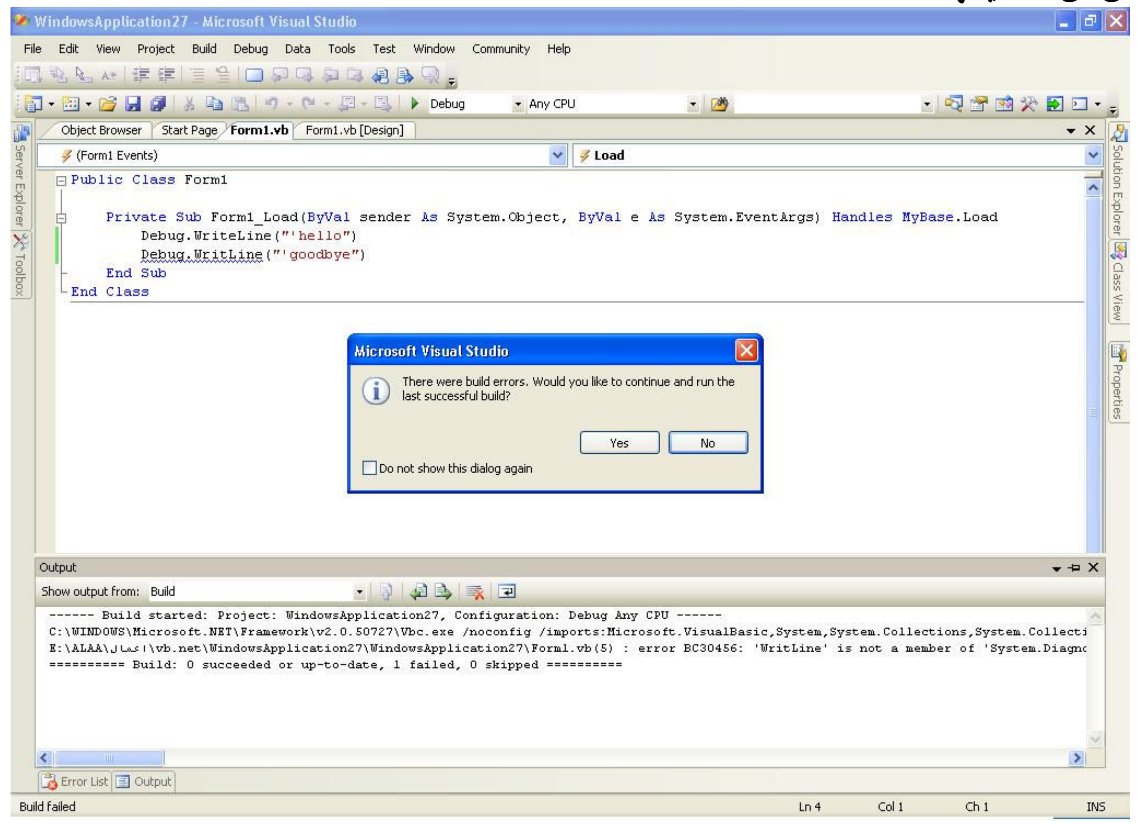
مثال :

اتبع الخطوات التاليه لطباعة العبارتين "Hello" "Goodbye" على شاشة الحاسب :

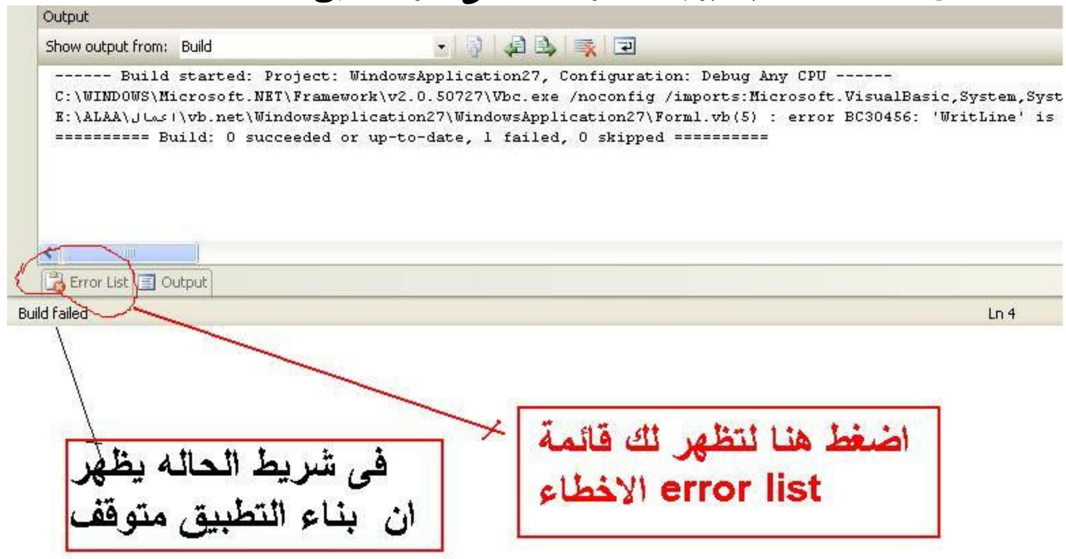
اكتب هذين الامرين داخل الحدث Load_Form1

```
Debug.Writeline("Hello")
Debug.Writline("Goodbye")
```

عند تشغيلك للبرنامج الان تجد ان البرنامج توقف واظهر لك رساله كما فى الصوره التاليه هذه الرساله تفيد بوجود خطأ داخل الاجراء لتتمكن من تصحيحها



انقر الزر No ليقوم الفيجوال بيسيك بايقاف تشغيل التطبيق وتظهر لك قائمه بالاخطاء داخل لوحة المهام Task List مع وصف مختصر لكل خطأ من هذه الاخطاء حيث يوجد خطأ واحد فقط فى الكود السابق .



ومن القائمه Error List اضغط على الخطأ نقرا مزدوجا تلاحظ ظهور شريط مضاء فوق الخطأ داخل نافذة الكود ولعلاج هذا الخطأ قم بتعديل كلمة `Writline` الى `Writeline` ثم اعد تنفيذ البرنامج مره اخرى
تظهر لك العبارتين التاليتين داخل نافذة الاخراج Output Window :

Hello
Goodbye

وبذلك فان الفيجوال بيسيك تعرف على الخطا واظهر رساله بالاخطاء

كيفية تجنب الاخطاء الهجائيه:

فالوقايه خير من العلاج فمن المزايا الرائعه Visual Studio.Net استخدام ما يسمى بالتكمله التلقائيه للكود Auto Code Completion من خلال السرد التلقائى للعناصر المتوقعه Auto List Members ،
مثلا لو كتبت اسم الكائن ثم نقطه سيدرك الفيجوال بيسيك انك ستكتب احد خصائص او وظائف الكائن ومن ثم يظهر قائمه بالخصائص والوظائف وبالنقر المزدوج او الضغط على المسطره بالكايبورد (Space) على احدى هذه الخصائص فان الفيجوال بيسيك يتولى امر كتابة الكود بدلا منك .
ومن الاشياء التى يتكفل بها الفيجوال بيسيك عملية ضبط الاحرف Capitalization للكلمات المحجوزه ولاسماء المتغيرات مما يعطيك انطباع مباشر عن كتابتك الصحيحه للكلمات .

ثانيا : اخطاء اثناء التشغيل

وهى اخطاء تحدث اثناء تشغيل البرنامج وتحدث نتيجة عدم قدرة الفيجوال بيسيك على التعرف عليها وبذلك يعجز عن تنفيذها ومن امثلة هذا النوع من الاخطاء القسمه على الصفر مثلا ، فقد يقوم المستخدم بادخال قيمه رقميه صفر عند القيام بعمليات ادخال مرتبطه بعمليات حسابيه ، لذالك لا بد من علاج هذه المشكله عن طريق عدم جعل القيم المدخله تساوى صفر .
احيانا يكون سبب الخطأ سهل الاكتشاف واحيانا يكون صعب ويحتاج الى خبره وتعمق فى تعقب الاخطاء فى البرامج لان الخطأ فى هذه الحاله يكون فى سطر اخر من الكود غير الذى حدده الفيجوال بيسيك .

ثالثا : اخطاء منطقيه

فهى اخطاء ليست اخطاء نحويه فى تعليمات البرنامج وليست اخطاء تظهر فى مرحلة التنفيذ ولكن الخطأ هنا من نوع جديد وهو الناتج المطلوب من البرنامج غير متوقع او غير منطقى ، فمثلا عند قيامك بعمليه حسابيه يقوم فيها البرنامج بحساب تكلفة شراء عدد خمسة حواسب اليه تكلفة الوحده منها او تكلفة كل جهاز على حده ١٠٠٠ ج مثلا . وان البرنامج اظهر ان التكلفه تساوى خمسه جنيه اجمالى تكلفة لخمسة اجهزه ،!!!!!
فهذه النتيجه غير منطقيه لان التكلفه تساوى ٥٠٠٠ جنيه وليست خمسه ،وفى هذه الحاله لا يقوم الفيجوال بيسيك بتصحيح او حتى تعقب الخطأ ، اذا لابد وان تقرأ الكود جيدا لتتعرف على الخطأ الحسابى وتعديله

كيفية تجنب الخطأ المنطقى :

١. بعد انتهائك من كتابة تعليمات البرنامج لا تتسرع بتنفيذ البرنامج قبل ان تقوم بمراجعة هذه التعليمات .
٢. لا تستخدم اسماء متغيرات متشابهه .
٣. لا تتسرع بتعريف كل متغيراتك على انها متغيرات عامه .
٤. تأكد من ان جميع تعليمات واجراءات وظائف البرنامج تعمل كما هو متوقع ومنطقى لتصميم البرنامج .

تصدير المشروع (ملف تنفيذى)

يحتوى اى مشروع يتم تنفيذه داخل Vs.Net على اصدارين الاول وهو اصدار النشر Release Version ويتميز بسرعة تنفيذ الكود .
والثانى هو اصدار التصحيح Debug او مايطلق عليه طور التصحيح .

WindowsApplication22 - Microsoft Visual Studio

File Edit View Project Build Debug Data Tools Test Window Community Help

Release | Any CPU

Debug
Release
Configuration Manager...

Form1.vb | Form1.vb [Design] | Start Page | Object Browser

lista | ItemCheck

انواع اصدار البرامج

```
        w.Kill()

    End Sub

    Private Sub lista_ItemCheck(ByVal [illegible] As Object, ByVal e As System.Windows.Forms.ItemCheckEventArgs
        prochandl = lista.Items(e.Index).SubItems(1).Text
        MsgBox(prochandl)
    End Sub

    Private Sub Form1_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
        Debug.WriteLine("hello")
        Debug.WriteLine("goodbye")
    End Sub
```

١ – اصدار النشر Release Version :

هو اصدار من البرنامج او التطبيق الجاهز للتوزيع لذا فهذا هو الاصدار الاخير من البرنامج والذى تتكاتف فيه جميع العوامل للحصول على اقصى تحسين فى البرنامج من حيث قلة الحجم وسرعة التفيذ .
فهذا الاصدار لايحتوى على اى معلومات خاصه بعمليات تتبع الاخطاء او تصحيحها لذا لا يمكن اكتشاف اى اخطاء بالبرنامج من خلال هذا الاصدار ومن ثم لايمكن تصحيحها .

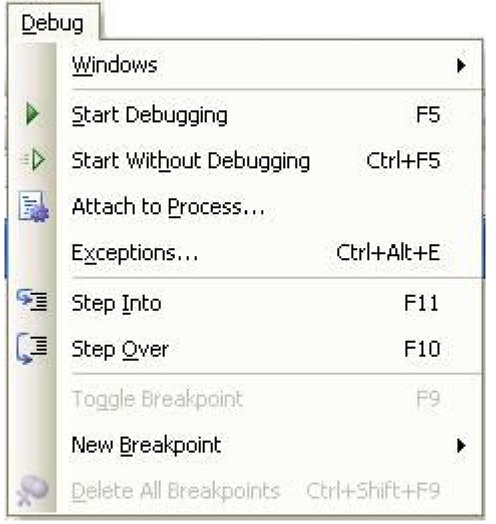
٢- اصدار التصحيح Debug :

فهوالاصدار المستخدم لتعقب الاخطاء فى البرنامج وتصحيحها لذا فهو لا يستخدم اى من تقنيات التحسين لان ذالك سيؤدى بالطبع الى تعقيد الكود المستخدم بدلا من تسهيله وبدلا من ذالك فهذا الاصدار يحتوى على جميع معلومات التصحيح الغير موجوده اساسا داخل طور النشر او اصدار النشر .
اعتقد انك الان عرفت الفرق بين النسخه Release و النسخه Debug .
☺ ☺

وهذه الاصدارات او الملفات من النوع EXE ويمكن استخدامها كما هى عند المستخدم ولكن لتقليل حجم البرنامج لابد من تحزيم البرنامج فى ملف SET UP وذالك عن طريق احدى برامج التحزيم ويمكنك ان تتعلم المزيد عن تحزيم البرامج من خلال منتدى المحيط العربى ابحث عنه وادخل قسم الدوت نت ستجد داخل قسم الدوت نت قسم اخر يحتوى على جميع المداخلات والمشاركات الخاصه بعملية التحزيم وحتى نشر البرامج.

اجراء عمليات التعقب والتصحيح

يمكنك البدا بعملية تعقب الاخطاء سطر بسطر وتصحيحها باختيار احدى الخيارات الموجوده بقائمة **Debug** كما يلى :

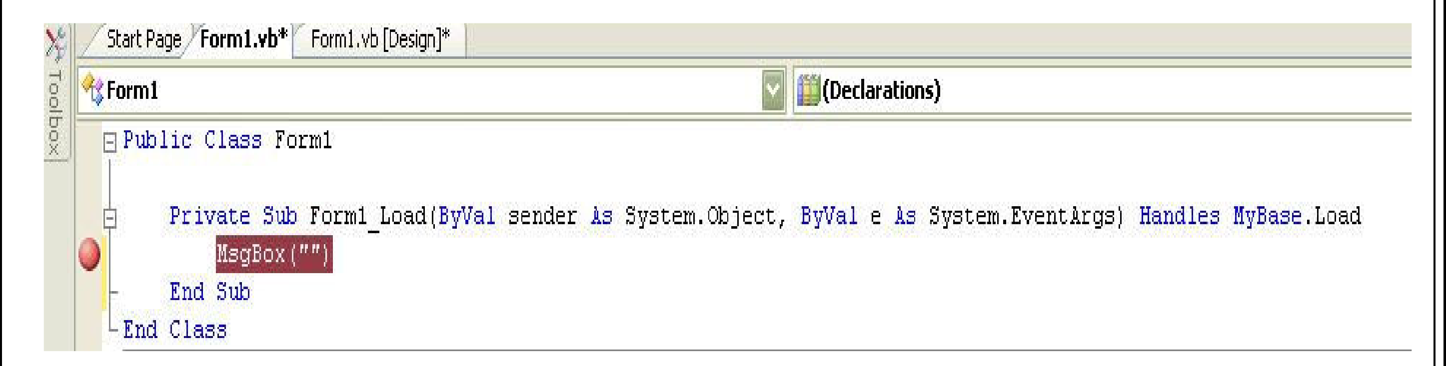


- يتسبب الاختيار **Start Debugging** فى بدا تنفيذ التطبيق او البرنامج الى ان يجد نقطه توقف **Breakpoint** وسوف نتعرض لنقاط التوقف بعد قليل .
- يتسبب الخيار **Start Without Debugging** فى تنفيذ البرنامج دون اعتبار لعمليات التنقيح او التعرف على الاخطاء وهو بذالك يستخدم حينما نكون على ثقه من صحة الكود المستخدم ونرغب فى سرعة تنفيذ البرنامج .
- يتسبب الاختيار **Step Into** فى تنفيذ البرنامج للسطر الاول داخل الكود ثم التوقف فاذا كان هذا السطر عباره عن استدعاء لاحدى الدوال يتم توجيه التحكم مباشرة الى داخل الداله المستدعاه ثم التوقف داخل هذه الداله قبل تنفيذ السطر الاول بها .
- يتسبب الخيار **Step Over** مثل الخيار السابق فى تنفيذ البرنامج للسطر الاول داخل الكود ثم يتوقف فاذا احتوى السطر الاول على استدعاء لاحدى الدوال يتم توجيه التحكم الى داخل الداله ولكن فى هذه المره يتم تنفيذ كود الداله بالكامل ثم العوده مره اخرى الى السطرالتالى لسطر استدعاء الداله ويتوقف تنفيذ البرنامج عند هذه النقطه .
- يتسبب الخيار **Step Out** فى تغيير التحكم من داخل الداله المستدعاه الى الداله التى قامت باستدعائها وهذا الخيار يستخدم حينما يكون التحكم داخل الداله المستدعاه حيث يتم تنفيذ الداله المستدعاه اولا قبل الانتقال الى الداله التى قامت باستدعائها
- يتسبب الخيار **Run To Cursor** فى بدا تنفيذ البرنامج حتى يتم الوصول الى اقرب نقطة توقف او الوصول الى مكان المؤشر
داخل الكود .
- يتسبب الخيار **Run To A Specified Function** فى بدا تنفيذ البرنامج حتى يتم الوصول الى داله معينه داخل الكود وحينئذ تتوقف عملية التنفيذ .

ملحوظه : بعض هذه الخيارات لاتكون متاحه اثناء التصميم وانما يمكنك مشاهدتها اثناء التوقف .

نقاط التوقف Break Point

كما ذكرت من قبل ان نقطة التوقف Breakpoint عباره عن مكان معين داخل الكود يتوقف عنده تنفيذ البرنامج .



```
Public Class Form1

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        MsgBox("")
    End Sub
End Class
```

انواع نقاط التوقف :

نقطة توقف الملف File Breakpoint : يتم تعيينها ى مكان ما داخل الملف او الكود او محرر الكود حيث يتوقف البرنامج عند الوصول الى هذه النقطه .

نقطة توقف الداله Function Breakpoint : ويتم تعيينها لداله معينه حيث يتوقف البرنامج عند الوصول الى هذه الداله .

نقطة توقف البيانات Data Breakpoint : ويتم تعيينها لقيمة احد المتغيرات وبالتالى يتم ايقاف تنفيذ البرنامج بمجرد تغيير قيمة هذا المتغير حيث يمكنك تعيين هذا النوع من نقاط التوقف داخل الكود القومى فقط .

نقطة توقف العنوان Address Breakpoint : ويتم تعيينها بعنوان او مكان معين داخل الذاكره ويتوقف البرنامج بمجرد الوصول الى هذا المكان .

ويمكنك تعديل سلوك نقاط التوقف عن طريق اضافة بعض الخصائص اليها :

Hit Count

Condition

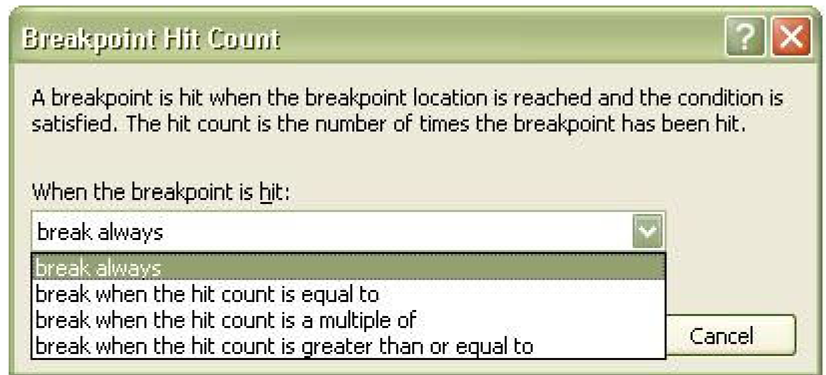
الخاصيه Hit Count

تقوم هذه الخاصيه بحساب عدد المرات التى يتم الوصول الى نقطة توقف معينه قبل ان يقوم البرنامج بالتوقف بالفعل ، وفى حالة نقاط توقف البيانات Data Breakpoint فان هذه الخاصيه تقوم بحساب عدد المرات التى تغيرت فيها قيمة هذا المتغير .

وفى الحقيقه ان القيمه الافتراضيه لهذه الخاصيه تتيح توقف البرنامج عند كل نقطة توقف الا انك تستطيع تغيير هذه القيمه ولعمل ذالك :



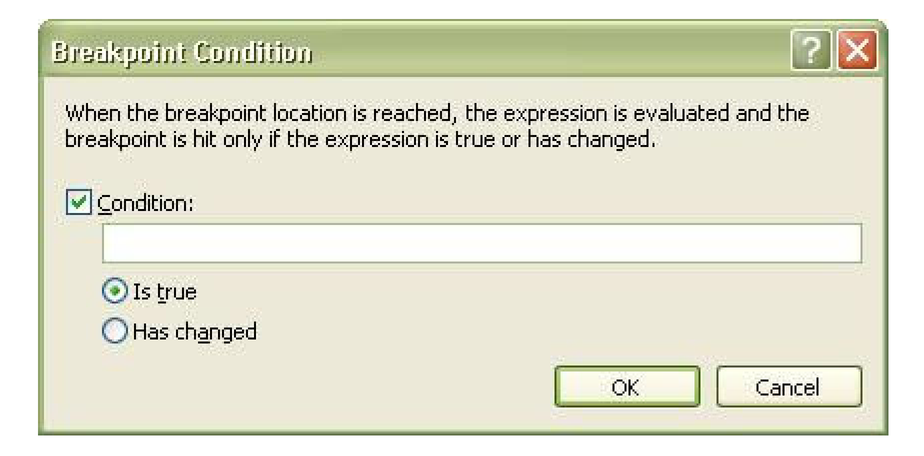
اضغط على الكره الحمراء التى ترمز الى وجود نقطة توقف عند هذا الكود بالماوس الايمن وستجد قائمة كما فى الصوره تحتوى على الخاصيه Hit Count و Condition اختار الخاصيه Hit Count وستظهر لك نافذه كما فى الصوره التاليه اختر منها مايروق لك :



ويمكنك ان تجرب وتتعرف على هذه الخاصيه فالوضع الاول Break Always هو الافتراضى وهو التوقف الدائم عند الوصول الى نقطة التوقف
اما البقيه فسوف ادعك تتعرف عليها بنفسك .

الخاصيه Condition

تقوم هذه الخاصيه بتعيين شرط معين يتم حسابه بمجرد الوصول الى نقطة التوقف ومن ثم يتم استخدام هذا الشرط لتحديد الوقوف عند هذه النقطه من عدمه فاذا تحقق الشرط يتوقف البرنامج عند هذه النقطه واذا لم يتحقق يتم تجاهلها والاستمرار فى تنفيذ البرنامج كانها غير موجوده :
فمن خلال نافذه حوار الخاصيه Condition كما فى الصوره التاليه يمكنك تعيين شرط نقطة التوقف :



قم بتعيين الشرط المطلوب داخل مربع النص Condition ثم قم بتنشيط الخيار Is True اذا اردت التوقف عند نقطة التوقف فى حالة صحة الشرط المحدد ،
اما الخيار Has Changed اذا اردت التوقف فى حالة تغير القيمه عن القيمه التى تم تحديدها .

ربط المصحح Debug بالبرامج الاخرى

يعتبر من احد السمات المميزه للدوت نت انك تستطيع ربط المصحح بالبرامج الاخرى ومن ثم تنفيذ عمليات التصحيح خارج الدوت نت حيث يمكنك استخدام هذه الميزه للاغراض التاليه :

- **يمكنك استخدام المصحح لتصحيح البرامج التى تم تنفيذها خارج الدوت نت .**
- **يمكنك تصحيح اكثر من برنامج فى نفس الوقت .**
- **يمكنك تصحيح البرامج الموجوده على الاجهزه البعيده مثال اجهزة الشبكات .**

اولا : تصحيح البرامج خارج الدوت نت (خارج البرنامج الذى تعمل عليه او تصنعه بنفسك)

كيف يتم ربط الاجهزه بالمصحح :

١ – من قائمة Debug اختر Processes او Attach To Process سيظهر لك فورم كما فى الصوره :

Attach to Process

Transport: Default

Qualifier: ALAA-2130BC0D38 Browse...

Transport Information

The default transport lets you select processes on this computer or a remote computer running the Microsoft Visual Studio Remote Debugging Monitor (MSVSMON.EXE).

Attach to: Automatic: Native code Select...

Available Processes

Process	ID	Title	Type	User Name	Session
ctfmon.exe	168		x86	ALAA-2130BC0D...	0
explorer.exe	1700	الاخطاء فى الدوت نت	x86	ALAA-2130BC0D...	0
IDMan.exe	248		x86	ALAA-2130BC0D...	0
IEMonitor.exe	2352		x86	ALAA-2130BC0D...	0
mspdbsrv.exe	3580		x86	ALAA-2130BC0D...	0
realsched.exe	2000		x86	ALAA-2130BC0D...	0
winamp.exe	2216		x86	ALAA-2130BC0D...	0
WindowsApplicatio...	3188		Managed	ALAA-2130BC0D...	0
WINWORD.EXE	2164	الاخطاء فى الدوت نت - Microsoft Word	x86	ALAA-2130BC0D...	0
wscntfy.exe	1936		x86	ALAA-2130BC0D...	0

Show processes from all users Show processes in all sessions Refresh

Attach Cancel

ستجد بداخل الفورم ليست فيو(List View) بها كل العمليات Process الموجوده على جهازك فى هذه اللحظه ، يمكنك اختيار اى من هذه العمليات لتكون قيد التصحيح فى حالة حداوث اخطاء .
وبعد قليل سنتعرف على كيفية اجراء عمليات التصحيح من هذه النافذه . ومن خلالها سنتعرف على نوافذ التصحيح .

ثانيا : تصحيح البرامج المتعدده

يمكنك تصحيح اكثر من برنامج بطريقتين مختلفتين :

- تصحيح البرنامج الاول ثم اختيار البرنامج الثانى من مستكشف الحلول Solution Explorer لتصحيح البرنامج الاول اختر البرنامج ثم اضغط بزر الماوس الايمن واختر Debug من القائمه الموضعيه .
- ابدا بتصحيح البرنامج الاول ثم استخدم الخطوات الموجوده بالبند السايق لربطط البرنامج الثانى بالمصحح وبالتالى يتم تنفيذ البرنامج الثانى خارج بيئة تطوير الدوت نت .

ثالثا : تصحيح البرامج الموجوده على الاجهزه البعيده

يمكنك من خلال المصحح ايضا تصحيح اخطاء البرامج الموجوده على الشبكه معك مثلا وذالك من خلال ربط الجهاز الموجود على الشبكه بالمصحح وتشغيل عملية اظهار المصحح اوتوماتيكيا حال حدوث اى خطأ بالبرنامج على الجهاز الاخر ويمكنك ربط الجهاز عن طريق الضغط على Browse الموجوده فى الفورم Attach To Processes كما فى الصوره السابقه .

نوافذ المصحح Debug

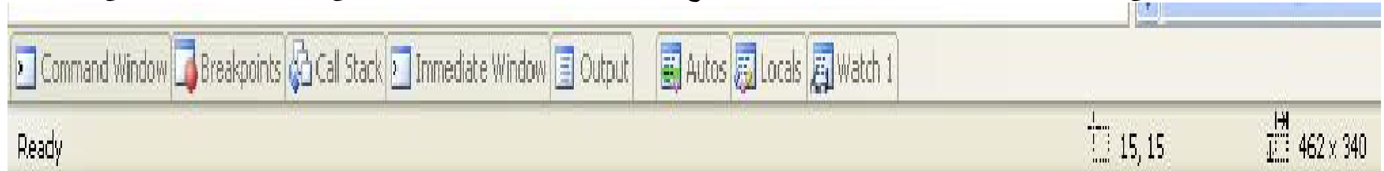
احدى الوظائف الاساسيه للمصحح تتمثل فى توفير معلومات عن البرنامج فى المراحل المختلفه لتنفيذه وهذه المعلومات غالبا ما تكون متاحه فى طور التوقف Break Mode وهو الوقت الذى يتم فيه تنفيذ البرنامج وابسط طرق الحصول على

المعلومه فى هذا الوقت تتمثل فى الاشاره الى الكائن الذى ترغب فى استرجاع معلوماته وليكن متغيرا او داله مثلا وذالك من النافذه الاساسيه Source Windows التى تحتوى على الملفات الاساسيه للبرنامج وحينئذ تظهر قيمة الكائن المختار داخل مربع نص صغير يسمى مربع المعلومات Information Box .

الطريقه الاخرى للحصول على معلومات البرامج هى استخدام نوافذ المصحح حيث تتيح لك الدوت نت عددا من النوافذ التى يمكنك استدعائها للحصول على المعلومات المختلفه عن البرنامج قيد الفحص او التصحيح :

النافذه Autos
النافذه Breakpoints
النافذه Call Stack
النافذه Disassembly
النافذه Locals
النافذه Memory
النافذه Modules
النافذه Registers
النافذه Running Documents
النافذه This
النافذه Threads
النافذه Watch

ويمكنك الوصول الى هذه النوافذ اثناء تشغيل عملية التصحيح من خلال الشريط الموجود اعلى شريط الحاله كما فى الصوره:



وفيما يلى نقوم بالقاء نظره خاطفه على كل هذه النوافذ :

النافذه Autos

يمكنك استخدام هذه النافذه فى طور التوقف فقط ،وتحتوى على معلومات تعكس قيم واسماء المتغيرات المستخدمه فى العباره الحاليه وكذالك العباره السابقه حيث تقوم النافذه بالتعرف على قيمة متغيرات العباره تلقائيا ومن هنا جاء اسم الننافذه .

النافذه Breakpoints

تقوم هذه النافذه بعرض جميع نقاط التوقف بالبرنامج والشروط المصاحبه لنقاط التوقف ان وجدت كما تحتوى هذه النافذه على شريط ادوات يحتوى على العديد من الخيارا ت التى تمكنك من اضافة نقطة توقف جديده او تعديل نقطه موجوده او حذفها كما تقوم النافذه بعرض معلومات عن اسم الداله او الملف الذى يحتوى على نقطة توقف .

النافذه Call Stack

يمكمك استخدام هذه النافذه فى طور التوقف فقط وتحتوى على معلومات عن الدوال الموجوده فى الوقت الحالى داخل جزء الذاكره Stack وكذالك معاملات الداله وقيم هذه المعاملات .

النافذه Disassembly

تقوم هذه النافذه بعرض الكود التجميعى لكود المترجم بالاضافه الى عرض عنوان الذاكره لكل سطر من سطور الكود .

النافذه Locals

تحتوى هذه النافذه على معلومات عن المتغيرات المحليه الموجوده داخل سياق الكود الحالى حيث تقوم بعرض اسم التغير وقيمته وكذالك نوعه ، ويقصد بالسياق الحالى افتراضيا الداله التى تحتوى على موضع التنفيذ الحالى .

النافذه Memory

تعرض هذه النافذة معلومات اضافيه عن الذاكره المستخدمه من قبل الوحدات المختلفه المستخدمه داخل البرنامج وبخاصة المعلومات التى لا يمكن عرضها داخل النافذه Watch او النافذه Variables. وتتغير هذه الذاكره باستمرار اثناء تنفيذ البرنامج حيث يطلق على هذه التغيرات بالتغيرات الحيه .

النافذه Modules

تعرض هذه النافذه قائمه بجميع النماذج المستخدمه من قبل البرنامج قيد التنفيذ وتتكون هذه النماذج من ملفات Exe او ملفات Dll حيث يظهر اسم النموذج وعنوانه ومساره وكذلك ترتيب النماذج داخل البرنامج بالاضافه الى رقم الاصدار والمعرف الذى يخصصه البرنامج لكل نموذج من هذه النماذج .

النافذه Registers

تعرض هذه النافذه المعلومات الموجوه داخل المسجلات Registers كما يتم اجراء التغيرات الحيه اثناء تنفيذ البرنامج قيد المعالجه والتصحيح . مثل النافذه Memory تماما لذا فان محتويات هذه النافذه تتغير باستمرار تبعا لتغير محتويات المسجلات .

المسجل Registers :

عباره عن منطقة تخزين داخل المعالج تحتوى على البيانات التى سيتم معالجتها كما يحتوى على عنوان البيانات داخل الذاكره ، لذا فان قوة المعالج تعتمد بشكل كبير على عدد المسجلات الموجوده بداخله .

ويمكنك تغيير اى قيمه بداخل هذه النافذه اثناء تصحيح البرنامج ولكن يجب ان تتوخى الحذر وتحرى الدقه فى القيم المدخله حتى لا يؤثر على تنفيذ البرنامج .

نافذة Running Documents

عباره عن مجموعه من التعليمات التى يتم تنفيذها تلقائيا دون اى تداخل من المبرمج فهى تشبه الى حد كبير المختزل ماكرو Macro حيث تقوم هذه النافذه بعرض قائمه بالمستندات المحمله داخل البرنامج قيد التصحيح .

النافذه This

تعرض هذه النافذه قائمه بجميع المؤشرات This المستخدمه داخل البرنامج وكذالك الكائنات التى تشير اليها هذه المؤشرات بم فى ذالك قيم هذه الكائنات وانواعها .

النافذه Threads

:Threads

عباره عن مسار تنفيذى داخل البرنامج ، حيث يحتوى البرنامج افتراضيا على مجرى واحد فقط فأذا قام البرنامج باستخدام اكثر من مجرى تنفيذى يطلق عليه فى هذه الحاله برنامج متعدد الخيوط او المسارات Multi Threaded Program .

وعلى ارغم من ذالك يتم تنفيذ عمليه واحده او مسار واحد فقط ويسمى Active Thread او المسار النشط

ويمكنك استخدام النافذه Threads للتحكم فى الخيوط او المسارات الموجوده داخل البرنامج ، واذا اردت تنشيط احد الخيوط انقر عليها من النافذه بالماوس الايمن واختر Switch To Thread .

النافذه Watch

تمكنك هذه النافذه من كتابة اى تعبير وايجاد قيمته كما يمكنك تحرير هذا التعبير اذا احتوى على اسم مسجل او متغير مثلا بينما لا يمكنك بحال من الاحوال تغيير او تحرير التعبير الثابت .

ويمكنك التحرير فى هذه النافذه عند الضغط بالماوس الايسر دبل كليك داخل الخانه فى العمود Name

ثم ادخل الاسم او التعبير ثم اضغط انتر لتظهر لك قيمة التعبير او المتغير داخل الخانه المقابله فى العمود Value .

الى هنا اخوانى نصل الى نهاية الدرس بالتوفيق باذن الله



تجد هذا الكتاب فقط فى منتديات المحيط العربى حيث النقاش والمشاركات الفعاله والكتب التعليميه النادر وجودها باللغه العربيه (كتب تشفير برامج و كتب الجرافيك وكتب الداتاباس و الاكسيس وغيره من الكتب العلميه ووالتعليميه الاخرى ....) مجانا من اجل نصرة الاسلام والعالم العربى فى العالم .وبقلم ابرز مبرمجى العالم العربى ....
اذا كان لديك اى سؤال او تعقيب على كل مافى هذا الكتاب من خلال المنتدى

Http://Www.Arabmoheet.Net/Forum

او الايميل الشخصى

Alaa_Gomaa2009@Yahoo.Com